

Enseñanza de la programación en la formación docente: análisis de la experiencia de los participantes desde una propuesta de evaluación reflexiva

Natalia Monjelat^{1*} y Ricardo Pablo Salvador^{1,2}

¹ Instituto Rosario de Investigaciones en Ciencias de la Educación (IRICE.CONICET-UNR)

² Escuela Superior de Comercio "Libertador Gral. San Martín" (UNR)
monjelat@irice-conicet.gov.ar; ricardopablo1@gmail.com

Resumen. El presente trabajo expone una propuesta de evaluación reflexiva que permite identificar aspectos relacionados con el diseño y la programación de producciones multimedia en Scratch. Esta propuesta toma aportes de trabajos previos y ha sido implementada en los talleres de programación desarrollados en el marco de la "Especialización Docente de Nivel Superior en Didáctica de las Ciencias de la Computación: Aprendizaje y Enseñanza del Pensamiento Computacional y la Programación en el Nivel Primario", implementada en Rosario (Santa Fe, Argentina). El análisis preliminar revela que la propuesta ha permitido a los participantes identificar diferentes aspectos y dimensiones respecto de los procesos en juego al programar. Se espera que estos análisis aporten evidencia que permita delinear futuros trayectos de formación docente en programación, poniendo en valor el proceso y las experiencias de los participantes. Asimismo, se pone a disposición la propuesta de evaluación, que puede emplearse y adaptarse para evaluar procesos desarrollados con otras herramientas que puedan emplearse en la enseñanza de la programación.

Palabras clave: programación, formación docente, Scratch, evaluación.

1 Introducción

En los últimos años, tanto a nivel regional como internacional, se ha comenzado a debatir la introducción en la escuela de las Ciencias de la computación (CC), como por ejemplo la programación informática. Estos contenidos generalmente no se han considerado en los planes de estudio para la enseñanza primaria o secundaria [2] por lo cual la articulación con la formación docente es un punto crucial. En Argentina, se iniciaron acciones de formación docente en estos temas, dentro de las cuales se destacan los convenios entre Institutos de Formación Docente, Fundación Sadosky con su proyecto Program.ar y universidades nacionales, que dieron lugar a especializaciones docentes de nivel superior en didáctica de las CC [3]. En este contexto, el presente trabajo expone una propuesta de análisis de experiencias de formación docente en programación, tomando como caso de estudio los módulos de taller de programación que se desarrollaron en el marco de la Especialización Docente de Nivel Superior en Didáctica de las Ciencias de la Computación: Aprendizaje y Enseñanza del Pensamiento Computacional y la Programación en el Nivel Primario [5,6]

* Este trabajo ha sido financiado por ANPCyT-FonCyT PICT 2016-1530.

desarrollada en la ciudad de Rosario (Santa Fe, Argentina) entre septiembre de 2017 y julio de 2019.

2 La propuesta de evaluación: acompañamiento y reflexión frente a la programación

El enfoque de evaluación propuesto parte de la premisa de que lo que sabemos y qué tipo de problemas podemos resolver, está ampliamente determinado por los recursos y las herramientas que tengamos a nuestro alcance en una práctica social específica. Sin embargo, predomina un uso instintivo y no reflexivo de las herramientas usadas, que se vuelven invisibles para nosotros [1]. Es por esto que apuntar a una reflexión que ponga en evidencia los procesos que se han desarrollado al programar, puede aportar datos sobre la experiencia en sí misma y las potencialidades de la herramienta en una situación particular, en este caso, de Scratch para el aprendizaje de conceptos de programación. Brennan y Resnick [7] proponen el uso de entrevistas basadas en artefactos, escenarios de diseño y documentación del alumno para poder evaluar el desarrollo del pensamiento computacional a partir de la programación¹. Dichos protocolos han sido adaptados dando lugar a la propuesta que aquí se presenta, revisada, a partir de una primera experiencia piloto [8, 9]. Se trata de un trabajo final individual, con temática libre, a realizar empleando Scratch, que debe contar con secuencias, bucles, variables, manejo de eventos y mensajes, operadores lógicos, sensores, bloques propios sin y con parámetros (al menos uno de cada uno), todos contenidos trabajados en los talleres. El punto clave a destacar de la propuesta de evaluación fue la inclusión de una serie de preguntas que permitieran a los participantes reflexionar sobre diferentes dimensiones relacionadas con el proceso de creación y programación. Para ello y basándonos en los aportes y autores mencionados, se plantearon las siguientes consignas:

- 1) Describir brevemente el proyecto: incluyendo la motivación del mismo, las reglas del juego y una explicación de cómo se usa.
- 2) Descripción del proceso de creación del proyecto:
 - ¿Cómo lo crearon?, ¿qué diferentes cosas intentaron?, ¿qué revisiones realizaron al proyecto original?
 - ¿Cómo decidieron qué objetos utilizar y dónde debían ir? ¿cómo eligieron los programas y su organización dentro del proyecto?
 - Si se inspiraron en algún otro proyecto, mencionar qué ideas tomaron y para qué las usaron.
- 3) Descripción de un momento en que algo no haya funcionado como esperaban: ¿qué ocurrió? ¿Cómo investigaron el problema? ¿Pudieron solucionarlo? ¿Cómo?
- 4) ¿Cómo podrían mejorar su trabajo? ¿Hay puntos que quisieron desarrollar y no llegaron? ¿Por qué? ¿Dificultades? ¿comentarios finales?
- 5) ¿Qué conceptos resultaron más complicados de implementar? ¿Por qué?

¹ Para esta instancia se tomaron como referencias las propuestas para la evaluación disponibles en el siguiente enlace: <http://scratched.gse.harvard.edu/ct/assessing.html>

Como puede observarse, el tipo de preguntas apunta a recuperar el proceso realizado, desde las etapas iniciales vinculadas con el diseño de la producción, hasta cuestiones relacionadas con la tarea de programar en sí misma, buscando develar lo que aconteció y no puede ser observado si solo se cuenta con la producción final, con el resultado.

Por otro lado, con la finalidad de otorgar a los docentes de estrategias para poder luego programar sus propias producciones con mayor autonomía, se propuso un acompañamiento en la elaboración de lo que sería este trabajo final, planteando inicialmente y previo a su elaboración la presentación de un anteproyecto que incluyera:

- 1) Trama. Explicar brevemente de qué trata el juego.
- 2) Presentación de los personajes (objetos o sprites) y los escenarios.
- 3) Programación para cada personaje. Explicar qué hace cada personaje y cómo piensan utilizar los conceptos requeridos (secuencias, bucles, variables, eventos y mensajes, operadores lógicos, sensores, bloques propios con y sin parámetros, etc.) para programar cada uno.

A partir de la presentación de los anteproyectos, los participantes contaron con el acompañamiento de los docentes del módulo para ir puliendo diferentes aspectos relacionados con la creación y programación de sus producciones que les permitirían llegar a la instancia de evaluación final. Esta presentación inicial permitió contar con un esquema de acción lo más claro posible que guiara a los docentes, ya que como observamos en la experiencia piloto, aspectos como elegir un fondo o cambiar un disfraz a un personaje podían insumir mucho tiempo y no necesariamente permitir el abordaje profundo de los conceptos de la programación.

4 Resultados preliminares

Esta propuesta de evaluación reformulada a partir de los resultados de la experiencia piloto inicial fue implementada en el “Taller de Programación 2” (duración de 30 horas) que tuvo lugar en el segundo semestre de la especialización docente. Durante esta instancia se recolectaron 31 anteproyectos, 31 informes con respuestas a las consignas propuestas y 31 producciones realizadas en Scratch, que fueron sometidos a un análisis cualitativo y descriptivo. Los resultados del análisis preliminar revelan que los proyectos, en su mayor parte, fueron imaginados como recursos para trabajar contenidos curriculares. En las descripciones del proceso de creación se observa una comprensión del funcionamiento de Scratch, se comentan sensaciones, cambios estratégicos en medio de los proyectos y se destacan aprendizajes logrados durante el desarrollo; se reutilizan otros proyectos y hay testeos. Las dificultades abarcan desde no saber cómo hacer algo concreto con Scratch, hasta proyectos que no funcionan como se espera; en pocos casos se omiten problemas en tanto que el “envío de mensajes” y variables son los más frecuentes. Casi la totalidad de los alumnos propone mejoras en la dinámica del juego, la estética y la interactividad; también acerca de generar aleatoriedad, agregar niveles al juego y dar feedback al jugador. La principal fuente de ideas para mejorar los proyectos parece ser la exposición del proyecto ante los pares y los profesores. Los conceptos difíciles refieren al uso de “bloques propios” con y sin parámetros, variables, operadores lógicos y mensajes.

5 Discusión y conclusiones

Consideramos que conocer las dificultades de los participantes, los conceptos en los que tuvieron mayores dificultades de implementación, qué aspectos quisieron desarrollar sin éxito, las mejoras que incluirían en sus proyectos, así como el proceso de diseño y creación que cada participante siguió al programar sus producciones permitirá aportar evidencia para delinear futuros trayectos formativos en programación. Asimismo, aunque este estudio se realizó empleando una herramienta en particular, puede ser adaptado para evaluar el proceso de programación que se lleva adelante empleando otros entornos de programación, pudiendo luego comparar dichos resultados para poder conocer qué herramientas resultan más idóneas para la enseñanza de la programación y el desarrollo del llamado pensamiento computacional. El análisis en profundidad de los datos obtenidos, así como el cruce de información entre la experiencia piloto y los diferentes tipos de datos obtenidos permitirá avanzar en el estudio de la experiencia docente en tareas de programación. Se espera con ello aportar evidencia que permita delinear futuros trayectos de formación docente en estas temáticas.

Referencias

1. J. Wertsch, *Voices of the mind. A sociocultural approach to Mediated Action*. Cambridge, Mass: Harvard University Press, 1991.
2. F. Heintz, L. Mannila, and T. Farnqvist, "A review of models for introducing computational thinking, computer science and computing in K-12 education," En *Proceedings - Frontiers in Education Conference, FIE*, 2016, pp. 1–9.
3. M. Borchardt and I. Roggi, "Ciencias de la Computación en los Sistemas Educativos De América Latina," 2017.
4. N. Monjelat and P. San Martín, "Programar con Scratch en contextos educativos: ¿Asimilar directrices o co-construir Tecnologías para la Inclusión Social?," *Prax. Educ.*, vol. 20, no. 1, pp. 61–71, 2016.
5. A. Casali, D. Zanarini, N. Monjelat, and P. San Martín, "Formación docente en Ciencias de la Computación: experiencias de la primera Especialización para el nivel Primario de la Provincia de Santa Fe," En *CACIC 2019*, 2919.
6. A. Casali, D. Zanarini, N. Monjelat, and P. San Martín, "Teaching and Learning Computer Science for Primary School Teachers: an Argentine Experience," in *Proceedings XIII Latin American Conference on Learning Technologies LACLO 2018*, 2018, pp. 349–355.
7. K. Brennan and M. Resnick, "New frameworks for studying and assessing the development of computational thinking," en *Annual American Educational Research Association meeting*, 2012, pp. 1–25.
8. N. Monjelat and A. Lantz-Andersson, "Teachers' narrative of learning to program in a professional development effort and the relation to the rhetoric of computational thinking," *Educ. Inf. Technol.*, 2019.
9. N. Monjelat, "Enseñanza de la programación en la formación docente: pautas pedagógicas desde una experiencia con maestros de primaria," en *Anales de las 48 JAIIO Jornadas Argentinas de Informática*, 2019, pp. 1–13.